

## Refine Search

### Search Results -

Term	Documents
PRIORIT\$	0
PRIORIT	7
PRIORITAET	1
PRIORITARILY	2
PRIORITARY	14
PRIORITAT	1
PRIORITATEN	1
PRIORITATIZATION	1
PRIORITATSANMELDUNG	1
PRIORITATSCODIERTES	1
PRIORITATSSTEUERUNG	1
(L12 AND PRIORIT\$ ).USPT.	26

There are more results than shown above. [Click here to view the entire set.](#)

Database:

US Pre-Grant Publication Full-Text Database  
 US Patents Full-Text Database  
 US OCR Full-Text Database  
 EPO Abstracts Database  
 JPO Abstracts Database  
 Derwent World Patents Index  
 IBM Technical Disclosure Bulletins

Search:

L13

Refine Search

Recall Text



Clear

Interrupt

### Search History

DATE: Friday, November 12, 2004    [Printable Copy](#)    [Create Case](#)

Set Name Query

side by side

Hit Count Set Name

result set

DB=USPT; PLUR=YES; OP=ADJ

L13    L12 and priorit\$26    L13

<u>L12</u>	l1 and l2 and L11	28	<u>L12</u>
<u>L11</u>	718/103.ccls.	364	<u>L11</u>
<u>L10</u>	l7 and L9	11	<u>L10</u>
<u>L9</u>	(higher or lower) near3 priorit\$	16154	<u>L9</u>
<u>L8</u>	(higher or lower) near1 priorit\$	15108	<u>L8</u>
<u>L7</u>	l5 and L6	27	<u>L7</u>
<u>L6</u>	l2 near3 data\$1	621	<u>L6</u>
<u>L5</u>	L1 near3 interrupt\$	1781	<u>L5</u>
<u>L4</u>	L1 near3 interrut\$	0	<u>L4</u>
<u>L3</u>	L1 near3 (interrut\$)	0	<u>L3</u>
<u>L2</u>	non near1 (real time\$1)	3030	<u>L2</u>
<u>L1</u>	real time\$1	104476	<u>L1</u>

END OF SEARCH HISTORY



US Patent &amp; Trademark Office

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide



THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)
Terms used **real time interrupt** and **priority**Found **66,955** of **145,831**

Sort results by

[Save results to a Binder](#)[Try an Advanced Search](#)

Display results

[Search Tips](#)Try this search in [The ACM Guide](#)
☐ Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐**1** [A real time priority scheduler](#)


Karl Ramsay, Jon C. Strauss

January 1966 **Proceedings of the 1966 21st national conference**Full text available:  [pdf\(495.56 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The real time priority scheduler reported herein evolved from a design effort to provide software aids for handling the interrupts that occur in hybrid computation. Since these interrupts are both internal and external (i.e., generated internal and external respectively to the digital portion of the computer), hybrid computation is representative of the more general real time scheduling problem. Real time scheduling, as exhibited by the needs of hybrid simulation, is basically di ...

**2** [ARPS: a new real-time computer](#)

Kenneth J. Thurber

October 1976 **ACM SIGARCH Computer Architecture News**, Volume 5 Issue 4Full text available:  [pdf\(1.14 MB\)](#)Additional Information: [full citation](#), [references](#), [citations](#)**3** [Real-time computing with lock-free shared objects](#)

James H. Anderson, Srikanth Ramamurthy, Kevin Jeffay

May 1997 **ACM Transactions on Computer Systems (TOCS)**, Volume 15 Issue 2Full text available:  [pdf\(390.82 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This article considers the use of lock-free shared objects within hard real-time systems. As the name suggests, lock-free shared objects are distinguished by the fact that they are accessed without locking. As such, they do not give rise to priority inversions, a key advantage over conventional, lock-based object-sharing approaches. Despite this advantage, it is not immediately apparent that lock-free shared objects can be employed if tasks must adhere to strict timing constraints ...

**Keywords:** critical sections, deadline monotonic, earliest deadline first, hard real time, lock free, rate monotonic, scheduling, synchronization, wait free

**4**[Real time analysis and priority scheduler generation for hardware-software systems](#)

with a synthesized run-time system

Vincent J. Mooney, Giovanni De Micheli

November 1997 **Proceedings of the 1997 IEEE/ACM international conference on Computer-aided design**Full text available:  [pdf\(283.64 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#) [Publisher Site](#)

We present a tool, called Clara, that performs real-time analysis and priority assignment for software tasks in a mixed hardware-software system with a custom run-time scheduler. We start from a system described in tasks/threads consisting of hardware specified in Verilog and software specified in C. We obtain the worst case execution time for each individual task. Then, based on the control flow of the application, Clara uses a dynamic programming algorithm to automatically find optimal priori ...

**Keywords:** hardware-software codesign, real-time analysis, run-time scheduler, worst-case execution time, rtos

5 Embedded application design using a real-time OS

David Stepner, Nagarajan Rajan, David Hui

June 1999 **Proceedings of the 36th ACM/IEEE conference on Design automation**Full text available:  [pdf\(105.02 KB\)](#)Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)6 Performance of REAL/IX<sup>TM</sup> fully preemptive real time UNIX

Borko Furht, J. Parker, D. Grostick

October 1989 **ACM SIGOPS Operating Systems Review**, Volume 23 Issue 4Full text available:  [pdf\(376.86 KB\)](#)Additional Information: [full citation](#), [abstract](#), [index terms](#)

The UNIX operating system, developed by AT&T Bell Laboratories, has become a standard operating system gaining rapid acceptance because of its superior flexibility, portability, and a number of support tools to increase programmer productivity. However, UNIX was originally designed for multitasking and time-sharing, and therefore conventional UNIX does not have an adequate response time and data throughput needed to support real-time applications. Many attempts have been made to ...

7 Ada 9X and real-time systems

John B. Goodenough

December 1992 **Proceedings of the conference on TRI-Ada '92**Full text available:  [pdf\(505.49 KB\)](#)Additional Information: [full citation](#), [index terms](#)8 Real-time convergence of Ada and Java<sup>TM</sup>

Ben Brosgol, Brian Dobbing

September 2001 **ACM SIGAda Ada Letters , Proceedings of the 2001 annual ACM SIGAda international conference on Ada**, Volume XXI Issue 4Full text available:  [pdf\(191.98 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


Two independent recent efforts have defined extensions to the Java platform that intend to satisfy real-time requirements. This paper summarizes the major features of these efforts, compares them to each other and to Ada 95's Real-Time Annex, and argues that their convergence with Ada95 may serve to complement rather than compete with Ada in the real-time domain.

**Keywords:** Ada, Java, Real-Time, asynchrony, garbage collection, scheduling, threads

9 Real-time software engineering in Ada: observations and recommendations

M. Borger, M. Klein, R. Veltre

January 1989 **Proceedings of the conference on Tri-Ada '89: Ada technology in context: application, development, and deployment**


Full text available:  [pdf\(1.82 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)



10 Bringing real-time scheduling theory and practice closer for multimedia computing

R. Gopalakrishnan, Gurudatta M. Parulkar

May 1996 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1996 ACM SIGMETRICS international conference on Measurement and modeling of computer systems**, Volume 24 Issue 1

Full text available:  [pdf\(1.18 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)



This paper seeks to bridge the gap between theory and practice of real-time scheduling in the domain of high speed multimedia networking. We show that the strict preemptive nature of real-time scheduling leads to more context switching, and requires system calls for concurrency control. We present our scheduling scheme called rate-monotonic with delayed preemption (rmdp) and show how it reduces both these overheads. We then develop the analytical framework to analyze rmdp and o ...

11 Shielded CPUs: real-time performance in standard Linux

Steve Brosky

May 2004 **Linux Journal**, Volume 2004 Issue 121

Full text available:  [html\(18.78 KB\)](#) Additional Information: [full citation](#), [abstract](#)

Add another tool to the real-time toolbox—simply dedicate one processor to your most critical task.



12 Modeling and validation of the real-time Mach scheduler

Hiroshi Arakawa, Daniel I. Katcher, Jay K. Strosnider, Hideyuki Tokuda

June 1993 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1993 ACM SIGMETRICS conference on Measurement and modeling of computer systems**, Volume 21 Issue 1

Full text available:  [pdf\(1.03 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)



13 Sporadic tasks in hard real-time systems

R. K. Allen, A. Burns, A. J. Wellings

September 1995 **ACM SIGAda Ada Letters**, Volume XV Issue 5

Full text available:  [pdf\(288.93 KB\)](#) Additional Information: [full citation](#), [abstract](#), [index terms](#)



In this paper, we illustrate how systems containing hard real-time sporadic tasks can be analysed for their worst case behaviour. In order to undertake this schedulability analysis, it is necessary to define the minimum inter-arrival time and/or maximum arrival frequency of sporadic tasks. Furthermore, at run-time it is essential to ensure that sporadic tasks are not invoked more often than has been guaranteed by the analysis. We assume that sporadics are invoked by interrupts and that interrupt ...

**14 Reflection of developing user-level real-time thread packages**

Shuichi Oikawa, Hideyuki Tokuda

October 1995 **ACM SIGOPS Operating Systems Review**, Volume 29 Issue 4Full text available:  pdf(1.02 MB) Additional Information: [full citation](#), [abstract](#), [index terms](#)

In multimedia environments, applications use continuous-media data, such as video data and audio data, to provide users with a wide variety of interaction schemes with computers. Continuous-media applications require more efficient and flexible support from real-time threads than such research real-time systems. To fulfill such requirements, we have been developing user-level real-time threads, which are real-time threads implemented at the user level. We have implemented two user-level real-time ...

**15 An operational semantics of real time design language RT-CDL**

L. Y. H. Liu, R. K. Shyamasundar

April 1989 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 5th international workshop on Software specification and design**, Volume 14 Issue 3Full text available:  pdf(893.34 KB) Additional Information: [full citation](#), [references](#), [index terms](#)**16 A real-time/time-share computer in a research and development environment**

C. D. Longerot, J. E. Marceau

January 1971 **Proceedings of the 1971 26th annual conference**Full text available:  pdf(878.06 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

A centralized computer with high speed peripherals, mass storage and very flexible input/output ports provides eighteen remote laboratory terminals with real-time/time-share computer service. The EMR 6130 Computer with Sandia designed interfacing provides real-time response in research and development activities involving on-line data acquisition, analysis and display, and includes features which allow process control and equipment programming activities. The system supports a variety of co ...

**Keywords:** Centralized computer, Computer interfaces, Data acquisition/display, Data transmission, Equipment programming, Experiment interface, Experimenter data interaction, Real-time computer service, Real-time monitor, Remote laboratory terminals, Remote time-share service

**17 Synapse: a small and expressive object-based real-time programming language**

Michel de Champlain

May 1990 **ACM SIGPLAN Notices**, Volume 25 Issue 5Full text available:  pdf(609.56 KB) Additional Information: [full citation](#)**18 Thoth, a portable real-time operating system**

David R. Cheriton, Michael A. Malcolm, Lawrence S. Melen, Gary R. Sager

February 1979 **Communications of the ACM**, Volume 22 Issue 2Full text available:  pdf(1.23 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)


Thoth is a real-time operating system which is designed to be portable over a large set of machines. It is currently running on two minicomputers with quite different architectures. Both the system and application programs which use it are written in a high-level language. Because the system is implemented by the same software on different hardware, it has the same interface to user programs. Hence, application programs which use Thoth are highly portable. Thoth encourages structuring progr ...

**Keywords:** minicomputer, operating systems, portability, real time

### 19 Response times in level-structured systems

Paul K. Harter

August 1987 **ACM Transactions on Computer Systems (TOCS)**, Volume 5 Issue 3

Full text available:  pdf(1.34 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#), [review](#)

Real-time programs are among the most critical programs in use today, yet they are also among the worst understood and the most difficult to verify. Validation of real-time systems is nonetheless extremely important in view of the high costs associated with failure in typical application areas. We present here a method for deriving response-time properties in complex systems with a level structure based on priority. The method involves a level-by-level examination of the system, in which in ...

### 20 A priority MAC protocol to support real-time traffic in ad hoc networks

Jang-Ping Sheu, Chi-Hsun Liu, Shih-Lin Wu, Yu-Chee Tseng

January 2004 **Wireless Networks**, Volume 10 Issue 1

Full text available:  pdf(264.87 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Carrier sense multiple access and its variants have been widely used in mobile ad hoc networks. However, most existing access mechanisms cannot guarantee quality for real-time traffic. This paper presents a distributed medium access control protocol that provides multiple priority levels for stations to compete for the wireless channel. One common channel is assumed to be shared by all stations. Stations are assumed to be able to hear each other (i.e., the network is fully connected). The channel ...

**Keywords:** carrier sense multiple access (CSMA), medium access control (MAC), mobile ad hoc network (MANET), quality-of-service (QoS), wireless communications

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2004 ACM, Inc..

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)